

**WEST** Generate Collection  Print

L8: Entry 1 of 14

File: USPT

Nov 25, 2003

DOCUMENT-IDENTIFIER: US 6654751 B1

TITLE: Method and apparatus for a virus information patrol

Drawing Description Text (8):

FIGS. 5, 6A, 6B, and 7 illustrate various aspects of a search/query user interface to be generated by a computer executing one embodiment of the illustrated invention shown in FIG. 1;

Detailed Description Text (21):

As illustrated, at processing block 312, the VIP data presenter 300 receives a request from a VIP subscriber 108A, 108B, or 108C to search for virus descriptor data 130 on the VIP database 110. In one embodiment, the request comprises one or more search criteria and may further include a format preference to list or sort the virus descriptor data that satisfies the search criteria in a specified way, e.g. sorted by company name, virus name, or the time the virus was reported, or to report statistical data about the virus descriptor data in a bar graph, or the like. At processing block 314, the VIP data presenter 300 generates a VIP query 140 from the received search criteria, and applies the VIP query 140 against the VIP database 110 to determine whether there are virus descriptor data 130 that satisfy the search criteria. If not, the VIP data presenter 300 returns control to the VIP subscriber 108A, 108B, or 108C and prompts the user to re-enter different search criteria. When there is virus descriptor data 130 available on the VIP database 110 that meet the search criteria, the VIP data presenter 300, at processing block 316, compiles the available virus descriptor data 130 in preparation for presentation to the VIP subscriber 108A, 108B, or 108C. At processing block 318, the VIP data presenter 300 formats the compiled virus descriptor data in accordance with the formatting preference requested by the VIP subscriber 108A, 108B, or 108C, and at processing block 320, the VIP data presenter 300 displays the formatted data to the VIP subscriber 108A, 108B, or 108C.

Detailed Description Text (24):

For example, ROM 403 and/or mass storage device 404 may include VIP data collector 200 instructions for a receiver to receive the VIP configuration data that identifies a location of the VIP data repository, a monitor to monitor the identified VIP repository to discover the virus descriptor links, a selector to select the discovered virus descriptor links, a collector to update the VIP database with the virus descriptor data obtained using the virus descriptor links, and a notifier to send notifications to the user/subscriber of the VIP database that new virus descriptor data has been collected. The ROM 403 and/or mass storage device 404 may further include VIP data presenter 300 instructions including a receiver to receive a search criteria from a user, a query generator to generate a query from the search criteria, a query applicator to apply the query to a virus descriptor data repository, and a user interface to present the virus descriptor data from the VIP database.

Detailed Description Text (29):

FIGS. 5, 6A, 6B, 7, 8, 9A and 9B illustrate various aspects of a user interface to be generated by a computer executing one embodiment of the illustrated invention shown in FIG. 1. Specifically, the VIP data presenter 300 operates in conjunction with the graphics capability of the operating system of the computer, such as Microsoft's Windows or Windows CE operating systems, to generate the search/query interfaces illustrated in FIGS. 5, 6A, 6B, and 7, and the results user interfaces illustrated in FIGS. 8, 9A and 9B. It should be noted that the illustrated interfaces reflect just one embodiment of the present invention, and that other variations of the interfaces may be employed without departing from the scope of the invention.

Detailed Description Text (30):

In one embodiment, the VIP data presenter 300 generates a virus search engine interface

500 as illustrated in FIG. 5, with at least one input area 502 for a user to input a search criterion, and a search command button 504 to initiate the VIP query 140 to the VIP database 110. In one embodiment, the search criterion is a word or phrase, such as the phrase "Code Red." The VIP data presenter 300 receives the search criterion in the input area 502 and, upon activation of the search command button 504, activates a VIP query 140 that is applied to the VIP database 110 to determine whether there is virus descriptor data 130 that meets the search criterion, e.g. whether there are any occurrences of the phrase "Code Red" in the virus name, virus description, or virus link portions of the virus descriptor data 130. If not, the user is prompted to re-enter a different search criterion in input area 502. In one embodiment, an advanced search link 506 is also provided for a user to navigate to the advanced search/query interfaces illustrated in FIGS. 6A and 6B, as described below.

Detailed Description Text (31):

In one embodiment, the VIP data presenter 300 generates an advanced virus search engine interface 601 and 602 as illustrated in FIGS. 6A and 6B, respectively. The advanced virus search engine interface 601 is comprised of at least two or more input areas to input various types of search criteria. For example, in one embodiment, there is one input area 602 to input a word/phrase search criterion. For example, as shown in the illustrated embodiment, the user has entered the word "Melissa" to find all virus descriptor data 130 containing the word "Melissa." The advanced virus search engine interface 601 further comprises at least one more input area 604 to enter a date or date/range search criteria, and/or at least one more input area 606 to enter a company name search criterion. In one embodiment, at least one more input area 608 is also provided to indicate whether to include virus descriptor data 130 that has an unknown announced date, i.e. when it cannot be determined from the virus description information collected from the virus information web servers 102A, 102B, and 102C when the virus was first announced.

Detailed Description Text (32):

In one embodiment, the advanced virus search engine interface 601 is further comprised of at least one input area 610 to enter a formatting preference. In the illustrated example of FIG. 6A, the formatting preference is to sort the results of the query by a specified field of the virus descriptor data 130, such as the name of the company that reported the virus information. In one embodiment, at least one more input area 612 is provided to enter a formatting preference to further sort the results of the query by another field of the virus descriptor data 130, such as the time reported. It should be noted that other formatting preferences might be provided for on the advanced search engine interface 601 without departing from the scope of the invention. Similarly to the search interface 500, the advanced search interface 601 is provided with an input command button 614 to activate a complex VIP query 140 to the VIP database 110 to determine whether there is virus descriptor data 130 that meets the advanced search criteria.

Detailed Description Text (34):

FIG. 6B illustrates an alternate embodiment of an advanced search interface 602, which is similar to the advanced search interface 601, except that an alternate embodiment of an input area 604 to enter a date or date/range search criteria is shown. Specifically, the advanced search interface 602 comprises an input area 622 to indicate a number of days in the past within which to confine the complex VIP query 140 generated by the VIP data presenter 300.

Detailed Description Text (36):

FIG. 8 illustrates a virus detail interface 800 that the VIP data presenter 300 generates in response to activation of the detail links 618 and 708 in the result summaries 616 and 706 that were generated in response to the above-described VIP queries 140. In the illustrated embodiment, the virus detail interface 800 is actually a web page hosted by the company from whom the virus information was collected. As shown, the virus detail interface 800 comprises a display area for the virus name and risk assessment 802, a display area for various items of bibliographic virus information 804, such as the discovery date, origin, length, type, subtype, minimum DAT, minimum engine, the DAT release date, and description added date. In addition, the virus detail interface 800 further comprises a display area for a description menu 806, that provides hyperlinks to other areas of the virus detail interface 800, such as the virus characteristics 808, and other areas (not shown), such as the symptoms, method of infection, removal instructions, and variants/aliases. The information displayed in the virus name/risk assessment area 802, the bibliographic virus information area 804, and virus characteristics 808, and other areas for displaying symptoms, method of infection, removal instructions, and variants/aliases are obtained from the content of

the virus descriptor data 130 that met the search criteria entered in the various search interfaces 500, 601, 602, and 702, and may include the actual description of the virus, or news items about the virus, or whatever other information about the virus that the anti-virus company posted to their virus information repositories 106A, 106B, and 106C on their virus information web sites 102A, 102B, and 102C. It should be noted that several other variations in the virus detail interface 800 may be employed without departing from the scope of the invention.

Detailed Description Text (51):

Having described and illustrated the principles of the invention with reference to illustrated embodiments, it will be recognized that the illustrated embodiments can be modified in arrangement and detail without departing from such principles. For example, while the foregoing description focused on monitoring, collecting, and presenting virus information in the context of a system implemented on distributed interconnected network, it will be recognized that the above techniques and analyses can be applied to monitoring collecting, and presenting virus information data or even other types of data in other contexts such as an intranet or other closed system, and having comparable limitations.

**WEST** Generate Collection 

L9: Entry 5 of 6

File: USPT

Jun 27, 2000

DOCUMENT-IDENTIFIER: US 6081814 A

TITLE: Document reference environment manager

Drawing Description Text (16):

FIG. 15 is a schematic block diagram of one embodiment of a query generator module illustrating steps for creation of a query consistent with the invention; and

Drawing Description Text (17):

FIG. 16 is a schematic block diagram of one embodiment of a standard query data structure consistent with the invention and provided by the query generator of FIG. 15.

Detailed Description Text (6):

The docloc table 180 may correspond to, be, include, exclude, or exist in addition to, the reference environment of FIG. 3, with its reference list of FIG. 6. This is true also of the docloc object of FIGS. 4a and 13, with respect to the reference environment object of FIGS. 3 and 4B. Similarly it is true of the docloc catalog object of FIG. 4B, with respect to any reference catalog object of 4B. Thus these corresponding structures illustrate embodiments of each other. FIGS. 15-16 illustrate certain embodiments of a query generator and a standard query data structure to support a reference environment manager.

Detailed Description Text (34):

The browser 98 may include a search engine interface 112. In general, the search engine interface 112 may be a client portion 112 of an internetwork browser search engine 132. Alternatively, the search engine interface 112 may actually be a search engine in its entirety. In one presently preferred embodiment, the search engine interface 112 includes a query generator 114. The query generator 114 includes an executable for obtaining or selecting query operands and operators, and properly ordering such information into a query. Thus, the query generator 114 may provide a query.

Detailed Description Text (37):

In general, the query generator 114 may be used in conjunction with any suitable search engine. Accordingly, the query generator 114 may formulate queries for searching sites 54 over an internetwork 50 for documents 146, searching hit lists 96 for adding entries to a document list 94, searching catalogs 110 to identify entries 102 corresponding to particular reference environments 100, and searching reference environments 100 and document lists 94 for particular documents 146 identified therein.

Detailed Description Text (38):

The navigation module 116 may operate in conjunction with the search engine 112 and query generator 114. The distinction between any search engine (e.g. engine 132), and an interface adapted to engage a search engine, need not confuse the fact that the interface 112 may effectively represent and be called a search engine for the browser. The navigation module 116 is responsible for providing inputs and outputs between a user and the user station 80. For example, a tool palette 118 may provide a selection of buttons 120, (e.g. access to features 120, or tools 120) for operating on menus 122. The navigation module 116 may provide menus 122 containing entries 124 corresponding to operations, data, entries, or the like.

Detailed Description Text (81):

A query 192f may present a query from any search engine. For example, a query 192f may be used to activate a search engine interface 112, a query generator 114, a search engine 132, or directory services search engine 164. Thus, in general, a tool 192 may be mapped to be activated to support a corresponding executable. Tools 120 may be linked to work together.

Detailed Description Text (112):

A query generator 212 may include a link-related data inputs module 214 for managing inputs used to generate queries. A formulator module 216 may provide templates 192t, logical operators, dialogue boxes, or the like, supporting operation of data structures to develop a query or a query data structure based on user inputs.

Detailed Description Text (113):

Any user inputs and outputs may be displayed in a dialogue box created, managed, activated, or otherwise supported by a dialogue support 218. Similarly, any query generator 114, 212 may provide a standard query data structure 220 as an output.

Detailed Description Text (114):

The standard query data structure 220 may include search criteria as operands. The standard query data structure 220 may also include the operators and associated logical delimiters for querying.

Detailed Description Text (140):

A provide browser step 266 may include providing 266 a user interface 104 or navigation module 116 for querying or engaging the search engine 132. However, the provide step 266 may typically include providing 266 an entire browser 98 having a search engine interface 112 and query generator 114 to operate with a primary search engine 132.

Detailed Description Text (141):

A generate query step 268 may engage the query generator 114 resulting in a client query 140 provided to a computer 54 networked as, for example, the internetwork server 86. Note that in the Figures, an illustration of a step may also represent a data structure created by the step. For example, the step of generating 268 a query 140 may also represent, and be referred to as, a query 268 generated thereby.

Detailed Description Text (151):

Providing 298 a browser 98 may include interfacing a browser 98 with the directory services search engine 164. Similarly, generating 302 a query may rely on operators and operands defined by a primary search engine 164, or some other search engine 112. Note that a query generator 114 and a resolver 112, 132, 164 therefore need not be integrated nor separated as illustrated herein. Functionality may be provided in any suitable architecture, create and resolve a query.

Detailed Description Text (210):

Referring to FIG. 15, and also FIG. 3, a query generator 114, 212 may be implemented in a variety of configurations. For example, a browser 98 may provide a display window 126 and a tool palette 118 for accessing tools 120, discussed above with respect to FIGS. 3 and 5. Construction of a query 390 may be done within the bounds of predicate calculus and to any degree of transparency to a user. Common query generator details, beyond those discussed is herein, are available elsewhere in the art.

Detailed Description Text (211):

In one embodiment, the query generator 114, 212 may include an obtain inputs step 386 or executable 386 to obtain inputs relating to a document 146, 184 (e.g. reference). The obtain inputs step 386 or executable 386 may be programmed to be controlled in part or entirely by the tool palette 118. The palette 118 may be used to operate on entries 124 in menus 122 displayed by the browser 98 or another display mechanism. A user may thus identify information effective or likely to uniquely identify a document 146, 184 sought.

Detailed Description Text (212):

The provide operator step 388 may occur in response to the nature of the inputs available, or may be determined in advance. For example, operators may be provided that are arranged in a standard query format, a standard template, or the like. Thus, the available operators for a query generator 212 and standard query data structure 220 may be limited. Alternatively, user interaction for deterministic logic, or the like may be used to provide operators 388, operands 389, or both.

Detailed Description Text (213):

Operands may be provided in a provide operands step 389 or executable 389. For example, tokens, pointers, values, and, the like to be operated on by operators 388 may be provided in the provide operands step 389. Note that each of the steps 386-390 may be thought of as a step in the process executed by a query generator 114, 212, but may also be viewed as the data provided thereby or used therein.

Detailed Description Text (218):

For example, the operands 389 may be defined by the create step 390 to identify information such as a reference environment catalog 110 reference environment object 174, reference environment 100, distinguished name 166, docloc object 170, docloc object distinguished name 392, document name 394, reference type 396, and the like. The distinguished name 392 may be used to search using the directory services search engine 164 for a docloc object 170. An entry 182 identifying any document 146, 184 reflected in a docloc table 180 pointed to by that docloc object 170 may then be identified. Similarly, the inputs 386 for a query generator 114, 212 (see FIG. 15) may provide a document name 394 corresponding to a document 146, 184 sought.

**WEST** [Generate Collection](#) [Print](#)

L2: Entry 1 of 2

File: USPT

Apr 25, 2000

DOCUMENT-IDENTIFIER: US 6055516 A  
\*\* See image for Certificate of Correction \*\*  
TITLE: Electronic sourcing system

## CLAIMS:

21. An electronic sourcing system comprising:

a requisition module including data fields, user-generated criteria entered into at least one of said data fields to generate at least partial criteria corresponding to a desired item;

a catalog collection searching module, said searching module including a collection of catalogs of items stored in an electronic format, a catalog selection criteria used to select less than said entire collection, said searching module being used to generate additional search-module criteria for said data fields of said requisition module;

a multiple purchase order generation module, said purchase order generation module creating multiple purchase orders from a single requisition created with said user-generated criteria and said search-module criteria;

wherein each of at least two catalogs include a generally equivalent item from a different source, said requisition module working in combination with said catalog searching module to determine multiple sources for said item;

wherein said multiple sources is limited by said catalog searching module providing a match according to said user-generated criteria, said search-module criteria and a determination system that located items are generally equivalent; and

wherein said determination system includes a cross reference table matching an identification code from a first located item with a second identification code from a second located item.